

Article

A Sparse Multiclass Motor Imagery EEG Classification Using 1D-ConvResNet

Harshini Gangapuram *  and Vidya Manian 

Department of Bioengineering, University of Puerto Rico, Mayaguez, PR 00681-9000, USA

* Correspondence: harshini.gangapuram@upr.edu

Abstract: Multiclass motor imagery classification is essential for brain–computer interface systems such as prosthetic arms. The compressive sensing of EEG helps classify brain signals in real-time, which is necessary for a BCI system. However, compressive sensing is limited, despite its flexibility and data efficiency, because of its sparsity and high computational cost in reconstructing signals. Although the constraint of sparsity in compressive sensing has been addressed through neural networks, its signal reconstruction remains slow, and the computational cost increases to classify the signals further. Therefore, we propose a 1D-Convolutional Residual Network that classifies EEG features in the compressed (sparse) domain without reconstructing the signal. First, we extract only wavelet features (energy and entropy) from raw EEG epochs to construct a dictionary. Next, we classify the given test EEG data based on the sparse representation of the dictionary. The proposed method is computationally inexpensive, fast, and has high classification accuracy as it uses a single feature to classify without preprocessing. The proposed method is trained, validated, and tested using multiclass motor imagery data of 109 subjects from the PhysioNet database. The results demonstrate that the proposed method outperforms state-of-the-art classifiers with 96.6% accuracy.

Keywords: compressive sensing; sparse representation; motor imagery EEG classification; 1D-CNN; residual networks



Citation: Gangapuram, H.; Manian, V. A Sparse Multiclass Motor Imagery EEG Classification Using 1D-ConvResNet. *Signals* **2023**, *4*, 235–250. <https://doi.org/10.3390/signals4010013>

Academic Editors: Ran Xiao and Quanying Liu

Received: 1 February 2023

Revised: 15 February 2023

Accepted: 1 March 2023

Published: 14 March 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Compressive sensing (CS) is an imminent signal-processing algorithm in areas such as video and image processing, magnetic resonance imaging (MRI) acquisition and reconstruction, and electroencephalogram (EEG) monitoring [1]. In a CS framework, we represent signals using a few nonzero elements on a suitable basis. This process is known as sparse representation. Then, we reconstructed the sparse signals through nonlinear optimization from these few nonzero coefficients. A great deal of research has been conducted in the healthcare industry on applying the CS framework to evolving fields such as brain–computer interfaces (BCIs).

BCI systems intend to provide an alternative communication channel for the brain's responses involving the subject's voluntary adaptive control [2]. Electroencephalography (EEG) is globally accepted among practitioners in developing BCI systems because of its inexpensiveness, noninvasiveness, and effectiveness in capturing the electrical activity of the brain. In the motor imagery (MI) process, an individual imagines a task involving moving parts of the body without moving the body parts physically. When the user imagines moving a body part (for example, legs, hands), event-related synchronization (ERS) and event-related desynchronization (ERD) develop in the power of the oscillatory EEG [3]. Detection of the distinct variations in the association between ERS and ERD forms the basic principle of MI BCI. The BCI-based classification system involves the collection of EEG data acquisition while the subject performs the MI task, feature extraction, and classification of MI for its neuroprosthesis application. We used a multichannel EEG amplifier to acquire EEG and remove noise and other artifacts from the signal to improve the

signal-to-noise ratio. After preprocessing the signals, many feature extracting techniques, such as spatial and bandpass filtering, were used to extract the signals [4]. Later, these extracted features were used for training and testing the classification model. We have to extract these features in both the training and testing phases. The classification accuracies are compromised because of the non-stationarity of brain responses in MI-related tasks. Therefore, for a real-time BCI system, the input features must be invariant to data variations caused by the non-stationarity of the signals. Additionally, we have to develop classification algorithms that can track these changes [4].

The CS framework applied to a wearable EEG-based BCI [5] is energy-efficient and can perform real-time wireless data transmission, signal acquisition, and signal processing. In a CS-based BCI system, the signals are compressed as a linear superposition of a few coefficients at the transmission end. Following that, these sparse coefficients are used for signal reconstruction at the receiver end, followed by feature extraction and classification (see Figure 1).

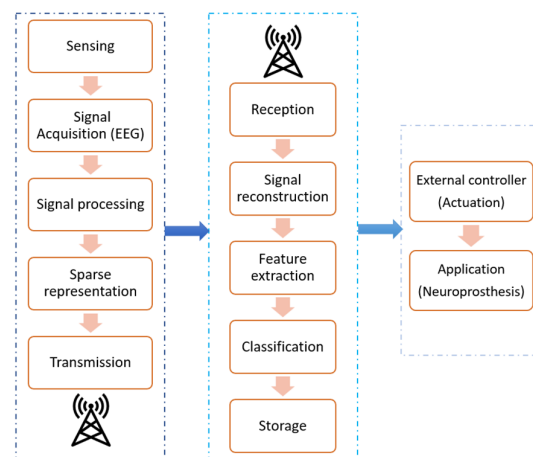


Figure 1. The compressive sensing system for EEG signals for BCI applications. In the transmission stage, the signals are represented sparsely. At the receiver end, the sparse signals are reconstructed, and features are extracted for classification and used for neuroprosthesis applications.

However, the reconstruction of the signal is computationally expensive and complex. A survey of recovery algorithms in compressive sensing is presented in [6], which mentions each algorithm's computational complexity in reconstructing the signal at the receiver end [1,7,8]. The classification of EEG signals is essential for an MI-based BCI system. Therefore, the sparsity-based classification system is efficient for an MI-based BCI system, eliminating the signal reconstruction and feature extraction at the receiver end [9]. This method reduces the memory required to store the reconstructed signal, lowers the computation cost, and develops a real-time prediction system. Researchers refer to such sparse representation-based classification (SRC) algorithms as 'compressive feature learning' algorithms [1]. First, the signals are represented as a sparse linear combination of extracted features from a dictionary, either predefined in the literature or learned from the data. Research is limited on dictionary learning and sparse representations compared with predefined dictionaries [10]. Next, state-of-the-art pattern recognition classifiers were used by researchers to classify these sparse features, such as support vector machines (SVMs) and linear discriminant analysis (LDA) classifiers [10–12]. However, such systems need two or more classifiers to address the multiclass classification problem and the computational cost and time increase with such classifiers [13]. The accuracy of 92% was achieved by [11] using four SVM classifiers in a one-vs-all scheme to classify four-class MI data.

In contrast, deep learning techniques such as convolutional neural networks (CNNs) can learn high-level features as the deep-stacked layers enhance the feature levels [14]. Many neural network models in the literature, such as [15–21], classify EEG signals. Further-

more, there is no need to stack two or more classifiers for a multiclass classification problem. However, poor classification accuracy was observed in the literature for sparsity-based classification using deep learning models.

Therefore, we introduce a novel sparse representation-based multiclass EEG classification method that reduces computational complexity and time while increasing classification accuracy. We extracted wavelet features (energy and entropy) directly from the raw EEG data, eliminating the preprocessing step. The length of the EEG epochs plays an essential role in improving the classification accuracy of the machine learning models. Therefore, we investigated different epoch lengths to identify which yields better classification accuracy. This paper aims to find an optimal epoch length for an MI-based classification while maintaining classification accuracy. In addition, we aimed to develop a framework that works efficiently for all users irrespective of their proficiency in MI-based BCI training. Therefore, we performed a subject-independent classification where we acquired training and testing data from different subjects. Finally, the proposed method retains the following contributions:

- Works well without preprocessing, such as using bandpass or spatial filters.
- Maintains the classification accuracy irrespective of the users' proficiency in performing MI tasks.
- The classification accuracy is improved with the selected channels related to the sensory motor cortex region.
- The accuracy is optimal with reduced epoch lengths and is therefore suitable for a real-time MI-based BCI.
- The proposed method is computationally inexpensive and faster than other machine learning algorithms.

The rest of the paper is organized as follows. Section 2 presents an overview of the methodology and the algorithms used in this manuscript, and Section 3 describes the classification results and compares the results with those of state-of-the-art methods. Section 4 presents a discussion and conclusions.

2. Materials and Methods

This paper proposes a sparse representation-based (SRC) EEG classification system using 1D-ConvResNet, as shown in Figure 2. We used the EEG data from the PhysioNet database [22]. We extracted EEG epochs (segments) and computed the feature vector. We noticed that each vector (feature) for a given trial has a dimension of epochs * channels. Next, we used k-fold crossvalidation to randomly split the feature vectors into training and testing samples. Next, we constructed the dictionary using the training data associated with feature vectors. Finally, we classified the sparse feature vectors directly without constructing a dictionary for the test data. A detailed description of the methodology is present in this section.

2.1. EEG Dataset

We used EEG recordings of 109 subjects performing four motor/imagery tasks using a 64-channel BCI2000 system to construct a general-purpose BCI system available in PhysioNet database [23]. Each subject performed 14 experimental runs containing two baseline runs and three two-minute runs of real/imaginary motor tasks. We used only the data corresponding to the MI tasks, as the BCI requires only imaginary movements for classification. The timing of the experiment is displayed in Figure 3. The experimental setup was as follows. The first two runs were the baseline runs, one with eyes open and the other closed. Next, the subject imagined opening/closing the left/right fist according to the displayed cue on the left/right side of the screen. If the visual cue appeared on the top of the screen, the subject was to imagine closing and opening both fists. If the visual cue was on the bottom of the screen, the subject was to imagine opening and closing both feet. The subject then relaxed after each cue.

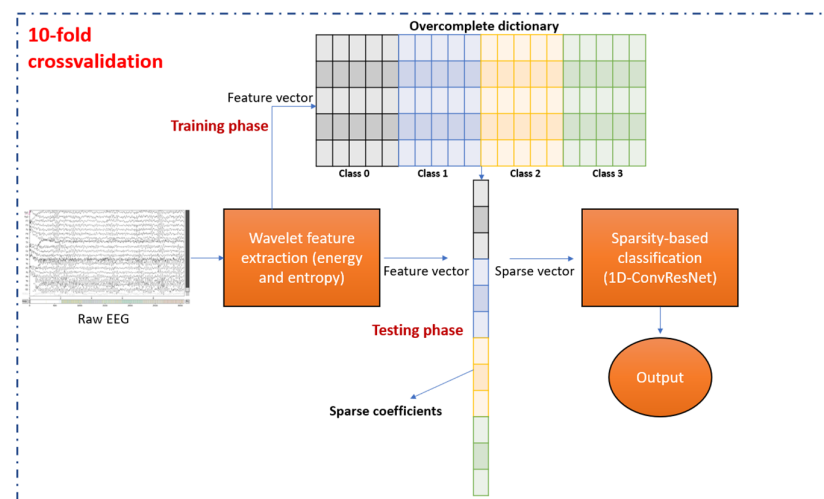


Figure 2. Overview of the methodology used in the present work.



Figure 3. Timing paradigm of the dataset.

2.2. Feature Extraction and Dictionary Construction

We took each trial of the raw EEG data ($t = 0$ s to $t = 6$ s) and extracted epochs every half second, as MI only lasts for half a second [10]. We extracted wavelet features (energy and entropy) using discrete wavelet transform (DWT) from each epoch, and for each class, as it produces a great distinction between the MI classes, which results in improved model accuracy [15–17,24]. In DWT, a signal $s[n]$ was passed through high-pass and low-pass filters to extract the respective frequency components. The low-pass filtered signal was downsampled at each level with a factor of 2 (see Figure 4) [18]. The coefficients from the low-pass filtered signal are known as approximate coefficients, and those from the high-pass filtered signal are referred to as detailed coefficients. We used the fourth-order Daubechies wavelet (db4) for wavelet feature extraction, as it is appropriate for analyzing EEG signals because of its smoothing [16]. Energy and entropy were calculated from the decomposed wavelet coefficients of the subbands and concatenated as dictionaries for each class. The training sample is the transpose of each epoch’s feature vector.

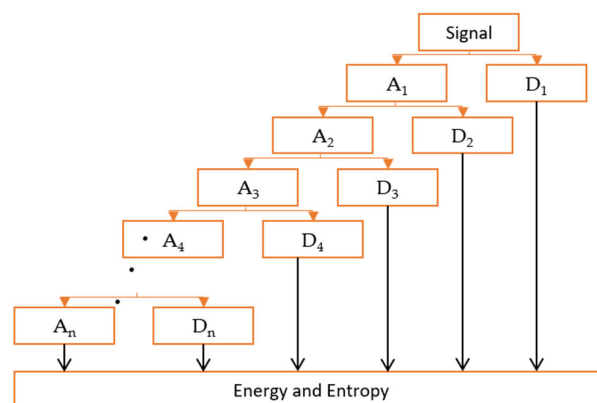


Figure 4. Wavelet feature extraction. A_1, A_2, \dots, A_n are approximate coefficients. D_1, D_2, \dots, D_n are detailed coefficients. We estimated the energy and entropy of all detailed coefficients and A_n .

We calculated wavelet dictionaries for each class separately and concatenated them into a single dictionary (D) as follows:

$$D = [D_1 || D_2, \dots || D_i] \in \mathbb{R}^{m \times n} \tag{1}$$

where D_1, D_2, \dots, D_i are the dictionaries of each class ($i = 0, 1, 2, 3$, as we have four classes), and D is overcomplete.

2.3. Sparse Representation

A sparse signal is a linear combination of a few coefficients selected from a prearranged overcomplete dictionary. The dictionary is constructed with the feature vector obtained from the wavelet features in D. According to CS theory, we can represent sparse signals using some basis. For a dictionary $D \in \mathbb{R}^{m \times n}$ and test data $y_i \in \mathbb{R}^m$ of class C_i (see Figure 5), we aimed to find the sparse feature (coefficient) vector $\alpha_i \in \mathbb{R}^n$ such that

$$y_i = D\alpha_i \tag{2}$$

which can be represented as

$$y_i = \alpha_{i,1}d_{i,1} + \alpha_{i,2}d_{i,2} + \dots + \alpha_{i,n_i}d_{i,n_i} \tag{3}$$

where sparse coefficient vector is defined by $\alpha = [0, 0, 0, \dots, \alpha_{i,1}, \alpha_{i,2}, \dots, \alpha_{i,n_i}, 0, 0, 0, \dots, 0]$, representing the nonzero coefficients for the i th class; thus, the corresponding EEG feature class can be determined. The given test feature vector is a linear superposition of the training sample feature vectors.

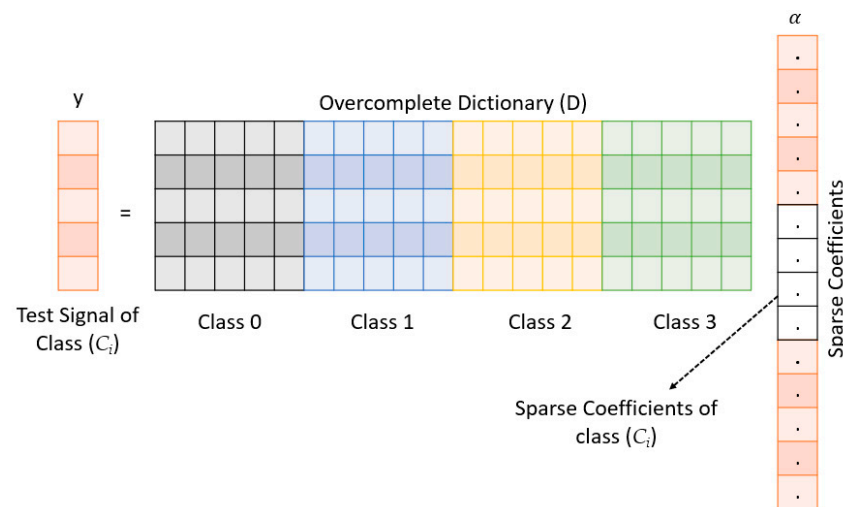


Figure 5. Sparse representation of the signal in compressive sensing.

The sparsest solution of $y_i = D\alpha_i$ can be obtained by minimizing the l_0 -norm given by Equation (4) as follows:

$$\min_{\alpha_i} \|\alpha_i\|_0 \text{ subject to } y_i = D\alpha_i \tag{4}$$

where $\|\cdot\|_0$ calculates the nonzero coefficients in α_i . The sparsest solution for the above equation is both NP-hard and numerically unstable because it requires all the possible combinations for the nonzero element positions in α_i , which is a large number. The literature provides many algorithms divided into convex relaxation methods and greedy approximation methods to solve the above equation. The convex relaxation methods include linear programming (LP), least absolute and shrinkage operator (LASSO), and iterative shrinkage and thresholding (ISTA); these methods relax l_0 -minimization to l_1 -minimization. Greedy approximation algorithms, such as orthogonal matching pursuit (OMP), give an

approximate optimal solution instead of solving the l_0 -minimization problem. In this paper, we performed the l_1 -minimization given in Equation (5):

$$\hat{\alpha} = \operatorname{argmin} \|\alpha_i\|_0 \text{ subject to } D\alpha_i = y_i, \quad (5)$$

In an ideal situation, Equation (5) should have nonzero elements corresponding to y_i . By analyzing nonzero coefficients in $\hat{\alpha}$, the i th class can be determined. While considering the limitations in modeling and noise, $\hat{\alpha}$ is not precisely zero, but it is close. Equation (6) was used to overcome this issue.

$$r_1(y) = \|y - D\delta_i(\alpha)\|_2 \quad (6)$$

We classified the test sample (sparse coefficients) using the above equation according to the residual approximations. The residual approximation must be sufficiently small to ensure that the test sample is close to the given class. By nullifying all the other coefficients corresponding to the other classes, i , $\delta_i(\alpha)$ is obtained, and we can classify class (i) by assessing the residuals as expressed in Equation (7).

$$\text{class}(y) = \operatorname{argmin}_i r_i(y) \quad (7)$$

2.4. Sparsity-Based Classification

2.4.1. Residual Network Theory

Deep CNN layers encounter vanishing or exploding gradient problems, as the back-propagation calculation of the gradient may lead to an exponential increase or decrease in the gradient with an increasing number of layers. Additionally, the training error might increase because of network degradation [25]. Furthermore, the model's learning rate decreases when there are several deep convolutional layers, and consequently, the accuracy is impacted. For CNN models, each layer's input is derived from the previous layer's output; hence, the model may collapse if it possesses many layers due to the calculation of gradients. Therefore, we introduce a residual network with a 'shortcut' structure [19]. The output of the middle layer was added to that of the first layer. The residual structure learns the difference between an optimal solution and a congruent mapping instead of just learning input-output mapping, as shown in Equation (8) [20,26].

$$F(y) = H(x) - x \quad (8)$$

where x is congruent mapping and $H(x)$ is the optimal solution.

Despite their advantages, traditional 2D-CNNs cannot be used to train and classify 1D sparse data. The signal should be represented in a 2D space to classify with 2D-CNNs. Therefore, we introduced a 1D-CNN to classify sparse features. A 1D residual learning block can efficiently overcome the performance degradation encountered with traditional 1D-CNNs [26]. The 1D-CNN reduces the features from the one-dimensional data sequence and maps its internal features. 1D-CNN helps analyze the sparse features of time series data, as the convolutional layer works on the fixed-length dataset, and the features can be located at any part of the segment. A kernel of a specified length is moved over the data by overlapping the layers to lose no features.

The architecture of the proposed 1D-ConvResNet model is shown in Figure 6, and each module proposed in the architecture is detailed in this section.

2.4.2. Input Module

The sparse EEG coefficients were taken as input to the 1D-convolutional layer. Here we used stride length as three instead of a pooling layer, as a large stride can result in adaptive learning of convolution kernel [26]. Next, we used a rectifier linear unit (ReLU) as an activation layer. The activation function either transforms the node's summed weight

into the node's activation or lets the input pass out of the layer. ReLU is a linear function that allows only positive outputs while thresholding the remaining outputs to zero [27].

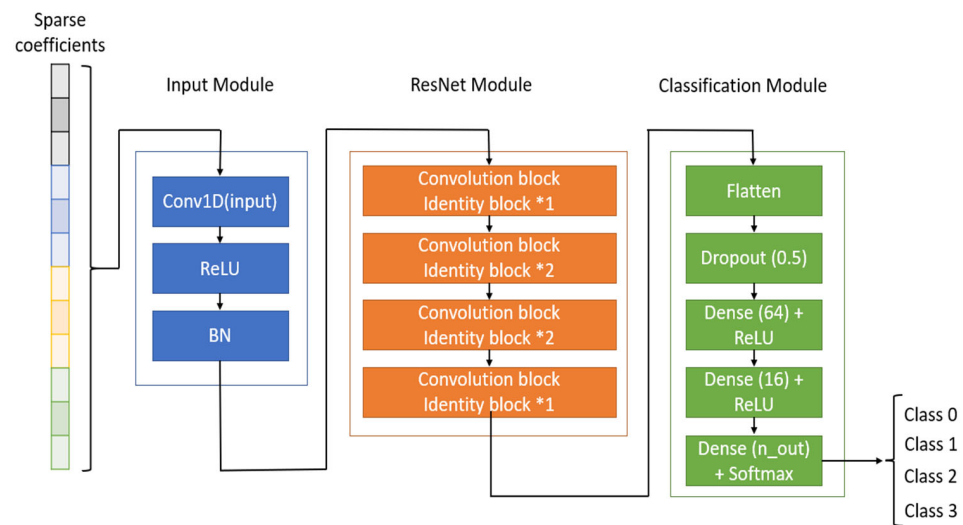


Figure 6. Architecture of 1D-ConvResNet. Three modules: Input module, ResNet module, and classification module. We performed a four-class classification of EEG sparse coefficients. (*) represents the number of blocks of each type.

While training a deep neural network (DNN) with many layers, the model might become susceptible to the structure of the learning algorithm and initial random weights. This is because the distribution of input layers may change when initial weights are updated after each minibatch. This disruption is known as ‘internal covariate shifting’. Batch normalization (BN) was used to normalize the inputs for each minibatch addressing internal covariate shifting [28]. Therefore, this procedure helps alleviate the learning process and reduces the number of training epochs in the neural network. The sparse EEG coefficients of each class are passed through the 1D-convolutional layer with an input shape $(N, 1)$. N is the number of sparse coefficients for each category.

2.4.3. Residual Learning Module

The residual module contains different combinations of layers in two blocks: identity and convolutional blocks [20] (see Figure 7). The output of the input module was taken as the input of the residual module. We divided the residual learning module into four stages, each consisting of a convolutional block and a series of identity blocks (see Figure 7). The number of filters for both blocks increases with the depth of the network to extract more relevant information required for the classification [19]. The convolution lock consists of a convolutional layer added to the shortcut structure.

2.4.4. Classification Module

The final module in the architecture (see Figure 6) is a classification module. In this module, we first flatten the entire pool of the feature map matrix obtained from the residual module to a single column and then feed it to the neural network. Then, we add a dropout layer to avoid overfitting of the residual network [25]. Next, the fully connected layers are activated by ReLU. The final layer is the classification layer, which contains the number of classes with a Softmax activation function, a probabilistic mathematical function that returns the index in a list that includes the most significant probability value. This function

maps the output of the fully connected layers (neurons) to values in the range of (0, 1) so that they sum to 1. Equation (9) expresses the Softmax function:

$$s(C_i) = \frac{e^{C_i}}{\sum_{j=1}^n e^{C_j}} \tag{9}$$

where $s(C_i)$ is the probability distribution of C belonging to the i th class, C_i is the input feature of the Softmax function, and j is the total number of classes corresponding to the different MI tasks. Hence, the Softmax function classifies the probability distribution estimate and reflects the class in the given list of targets that achieves the highest score for the given training/test sample.

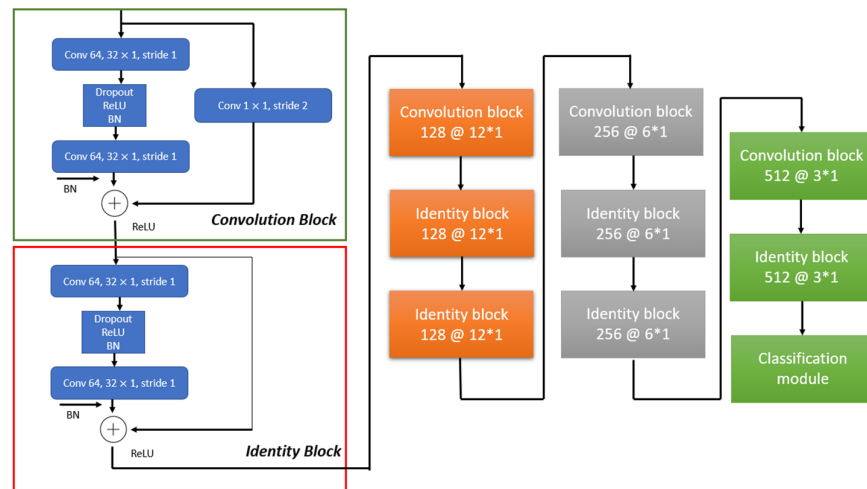


Figure 7. ResNet Module architecture. The ResNet module consists of a combination of Convolution block and Identity blocks. (*) represents the number of blocks of each type.

2.5. Subject-Independent Classification

BCI systems need large-scale MI data for training and classification. Many subject-specific classification models have been proposed in the literature for BCI systems [2,10,29–38]. However, individuals cannot imagine a motor task beyond a certain duration, and even across different sessions, the data collected from the same subject may present any external artifacts, leading to intrasubject variability [39]. Consequently, researchers have started collecting small samples from many subjects instead of collecting many samples from each subject in a single session [22,39]. A BCI system should perform subject-independent classification (see Figure 8) as it helps adapt to many people [40]. We used 82 subjects (80%) EEG data for training and validation and the remaining 21 subjects (20%) EEG data for testing. The sampling frequency of the data was 160 Hz. We excluded the data from 6 subjects because of the presence of mistrials, which would lead to discrepancies in the training, validation, and testing of the model [41]. Hence, subject-independent classification is performed on the EEG data from 103 subjects recorded during 4 MI tasks (left fist, right fist, both fists, and both feet). The subjects were split randomly for k-fold crossvalidation, where (k-1) folds are used for training, and the remaining fold was used for validation. Finally, we saved the model weights and tested the model with the remaining 20% of the samples from the subjects not known to the model.

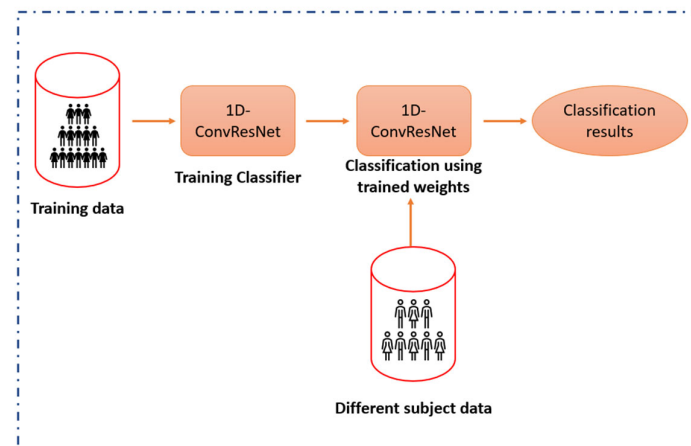


Figure 8. Subject-independent classification using transfer learning. We trained the model using training subjects' data. Later, we classified test subjects' data using the trained weights.

3. Results

We ran the model on a PC with an Intel® Core™ i7 processor, 16 GB RAM, NVIDIA GeForce RTX 2070 GPU with 8 GB memory, and software (Python 3.7 and TensorFlow 2.2).

3.1. Subject-Independent Classification

We performed the subject-independent classification where training, validation, and testing were performed on the MI data of different subjects (see Figure 9). Testing the model on data corresponding to different subjects from the trained data is challenging, as differences in the subjects' motor skills may result in variations in imagining the movements. The diagonal cells of the confusion matrix (Figure 9a) show that the model achieves 100% classification accuracy on the MI of the right fist. The areas under the ROC curves and the precision–recall curves also validate these results; the subject-independent classification accuracy is 96.6%.

3.2. Performance of the Proposed Method with Different Epoch Lengths

Each trial of the MI task in the dataset lasts 4 s, and any subject can perform the MI task many times during one trial. According to [10], one MI lasts for 0.5 s. Therefore, we used an initial length of the epoch as 0.5 s. However, it is essential to decrease the epoch length to reduce the computational time for a real-time BCI system. Therefore, we extracted different epoch lengths to investigate which epoch length yields a better classification accuracy and lesser computational time in training and testing. Table 1 presents the training and testing computational time and accuracy of different epoch lengths for the proposed method. It is evident that the training time increased with decreasing length of the epochs, and testing time was significantly reduced. However, classification accuracy was highly compromised for 0.2 s followed by 0.3 s. We can consider 0.4 s epoch length for optimal accuracy and computational time, but 0.5 s epochs have greater accuracy than other epoch lengths.

Table 1. Performance of the proposed method with different epoch lengths.

Epoch (Length of Trial in Seconds)	Accuracy (%)	Training Time (s)	Testing Time (s)
0.1	76.08 ± 0.45	300	0.25
0.2	62.59 ± 0.25	259.06	0.33
0.3	72.8 ± 0.24	140.8	0.39
0.4	90 ± 0.16	110.63	0.45
0.5	96.6 ± 0.18	98	0.5

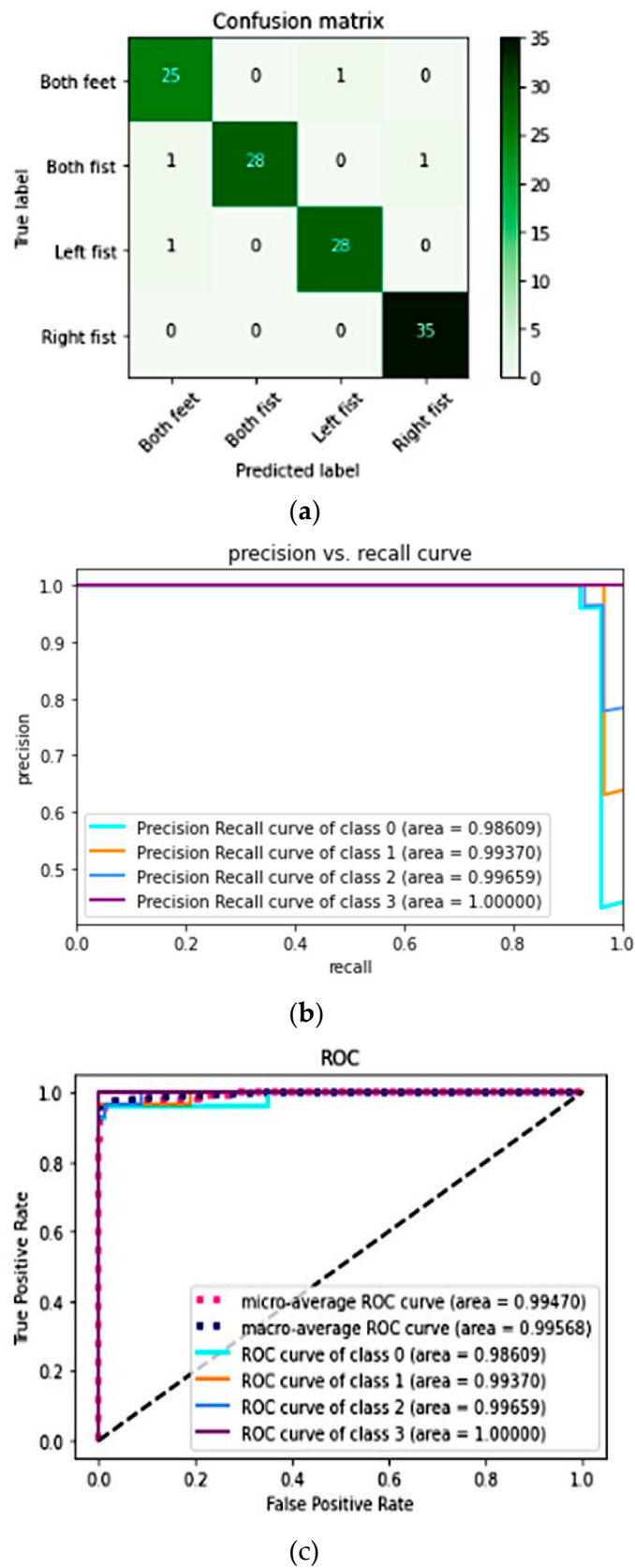
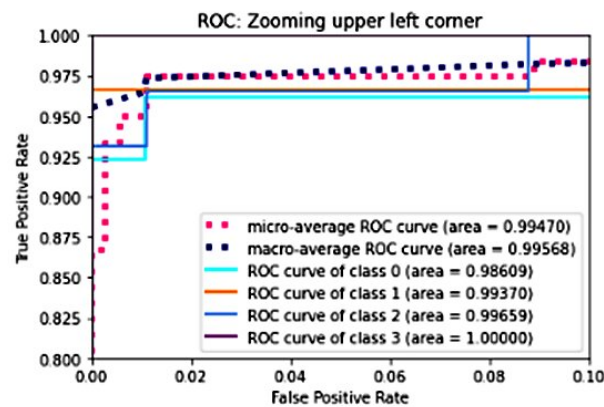


Figure 9. Cont.



(d)

Figure 9. Classification results for the subjects in the PhysioNet dataset (test data) for subject-independent classification. (a) Confusion matrix. The model achieves 100% accuracy for right fist MI. (b) Precision–recall curve. (c) ROC curves. (d) Zoomed-in view of the upper left corner of the ROC curve plot. Precision–recall curve and ROC plot also validate these results.

3.3. Performance of the Proposed Method with Different Epoch Lengths and Selected Channels

The EEG data of the dataset used in this manuscript is 64 channel EEG data. However, MI tasks only activate the sensorimotor areas for the MI task. Therefore, we selected the EEG channels corresponding to the motor cortex region. These include FC1, FC2, FC3, FC4, FC5, FC6, C1, C2, C3, C4, C5, C6, CP1, CP2, CP3, CP4, CP5, and CP6. We used different epoch lengths, and the selected channels applied the proposed methodology (see Table 2). There was a significant improvement in accuracy, training, and testing times with the selected channels.

Table 2. Performance of the proposed method with different epoch lengths and selected channels.

Epoch (Length of Trial in Seconds)	Accuracy (%)	Training Time (s)	Testing Time (s)
0.1	77.9 ± 0.18	250	0.18
0.2	63 ± 0.22	213.45	0.18
0.3	76.85 ± 0.18	125	0.2
0.4	94.8 ± 0.25	96.25	0.32
0.5	98.75 ± 0.46	75	0.33

3.4. Performance of the Proposed Method with Different Features

State-of-the-art machine learning algorithms have used various features in the wavelet domain, frequency domain, and time domain to classify MI EEG data. To investigate the efficiency of the proposed model, we extracted different features from the EEG epochs. However, constructing a single overcomplete dictionary based on these features takes more computational time. Therefore, we compared additional features individually using the proposed method for accuracy and computational time. The performance of the proposed method using different features is presented in Table 3. We used 0.5 s epoch lengths to compare as they yield better classification accuracy. In addition, the combined wavelet features with energy and entropy produced better classification accuracy than other methods.

3.5. Comparison of the Proposed Model with Different Classification Models

To investigate the competence of the proposed method, we compared it with four different classifiers: k-NN, LDA, SVM, and CNN (see Table 4). The first three classifiers are

binary classifiers, and hence we must combine the classifiers in a one-vs-one manner. CNN can be carried out with a multiclass classification. We used the Scikit Learn library in Python to classify using the first three classifiers. TensorFlow was used for classification using CNN. Next to the proposed model, SVM has better classification accuracy and computation time. However, the computational cost increases as we must carry out a series of binary classifications in a one-vs-one manner. Although CNN carries out a multiclass classification, its accuracy is significantly compromised as we tried to train and classify 1D-data with 2D-CNN.

Table 3. Performance of the proposed method with different features.

Features	Accuracy (%)	Training Time (s)	Testing Time (s)
Wavelet entropy	93 ± 0.28	110	0.34
Wavelet energy	93.9 ± 0.25	111	0.34
Kurtosis	86.11 ± 1.26	122.76	0.34
Wavelet entropy + Wavelet energy (Proposed)	96 ± 0.19	125.6	0.38
Power spectral density	91.2 ± 0.84	156	0.54
Common spatial pattern	94.5 ± 0.456	142	0.45

Table 4. Comparison of the proposed model with different classification models.

Classifier	Accuracy (%)	Testing Time (s)
SVM	92.46 ± 0.28	0.45
LDA	68 ± 2.98	0.78
k-NN	56.2 ± 2.74	0.76
CNN	66.9 ± 0.25	0.62
Proposed model	96.6 ± 0.18	0.38

3.6. Comparison with State-of-the-Art Methods

The subject-independent classification accuracy for the proposed model is 96.6%, outperforming all state-of-the-art methods (refer Table 5). Most existing methods are built on the EEGNet architecture [34], which was initially proposed to classify P300 data [35]. Hence, the classification accuracy is compromised for MI data. Furthermore, all previous state-of-the-art methods require more storage space and higher computational cost for the data than the proposed method, as they were not optimized for compressed features. The proposed method achieves higher subject-independent classification accuracy than the existing methods with less storage space and a computational time of 0.32 s, sufficient for real-time classification.

Table 5. Comparison with state-of-the-art methods.

Methodology	Accuracy (%)	k-Coefficient Values
Deep recurrent CNN [40]	70.64	-
EEGNet Fusion [42]	83.8	-
CNN [43]	65.73	-
CNN + electrode pair signals [44]	72.51	-
EEGNet [33]	65.07	-
Proposed method	96.6	0.95

4. Discussion

State-of-the-art methods use different machine learning, and deep learning algorithms carry out feature-based EEG classification, which is computationally expensive, and the accuracy is compromised. Furthermore, subject-independent classifications suffer from intersubject variability in performing the MI tasks. The proposed methodology overcame these limitations, and we discuss each scenario in detail in this section.

4.1. Accuracy

First, we extracted the wavelet features (energy and entropy) for the sparse representation of EEG data. The wavelet transform enables the signal decomposition into subsignals consisting of different frequencies (high and low) with distinct spatial and directional characteristics [45]. Therefore, the sparse representation of wavelet features (energy and entropy) provides the discriminable features necessary for EEG classification in the sparse domain [10]. Then, a sparse dictionary was constructed based on the wavelet features. The overcomplete dictionary D consists of four sub-dictionaries corresponding to each class of the data (refer to Equation (1)). Hence, there is a high variance between the sub-dictionaries, and each test signal can be represented through the sub-dictionary corresponding to the class. Therefore, there is no chance of misrepresentation of the class of the test signal, and the classification accuracy is significantly improved.

Furthermore, we have used different epoch lengths to investigate which epoch length yields better classification accuracy in two scenarios: without selecting channels and with selecting channels. The dataset contains 64-channel EEG data and MI tasks only activate the sensorimotor areas for the MI task. Hence, we selected the EEG channels corresponding to motor cortex region, and there was a significant improvement in training time and testing time improved with selected channels along with classification accuracy.

We have compared our methodology with state-of-the-art methods. The proposed methodology that outperformed other studies has two major drawbacks. Firstly, selection of features for sparse representation as the sparse features should be able to carry enough information for classification [40]. Second, the proposed classifiers were based on EEGNet architecture, which was initially proposed to classify P300 data [33].

4.2. Computational Time

The MI-based classification algorithms require heavy preprocessing of the data, such as artifact removal, filtering (bandpass), feature extraction, feature selection, and classification. Hence, the computational cost and time increase with such procedures. On the other hand, the proposed method extracts the wavelet features directly from the raw EEG without preprocessing, and an overcomplete dictionary was built through uncorrelated sub-matrices corresponding to four different classes. After constructing the overcomplete dictionary, the sparse coefficients were obtained and used for classification. The purpose of sparse representation is to form a redundant dictionary with few coefficients corresponding to each class of data for training the model. As a result, the testing signal requires only the time to calculate a single sparse feature for classification.

Furthermore, we used 1D-ConvResNet, which efficiently learns and classifies the sparse features and can be used for a multiclass classification instead of combining binary classifiers. Hence, the computational cost was significantly reduced compared to other machine learning algorithms.

4.3. Intersubject Variability

Large-scale MI data are needed to build a BCI system. Many subject-specific classification models have been proposed for BCI. One should note that the subject's performance in the MI task will degrade after a certain time. Hence, researchers often collect data across multiple sessions to minimize this performance degradation; nevertheless, intrasubject variability can still be observed [38]. In recent years, databases have been constructed with the MI data of many subjects with fewer subject-specific sessions to perform subject-

independent classification [22,38]. The subject variability issue in subject-independent classification is controversial; however, this issue is also present in subject-specific classifications. Recent studies have performed subject-independent classification with low accuracy (see Table 4). One reason is that variable subject-related artifacts, such as the subject's mental state, can affect the MI task data. Feature extraction methods such as wavelet features can effectively extract EEG band information related to MI tasks (8–30 Hz) with less computational complexity. This proposed methodology has addressed intersubject variability issues by extracting maximum signal information associated with MI tasks and representing its corresponding class efficiently.

4.4. Limitations of Validation

We have carried out a subject-independent classification, and the intersubject variability threats can still be debated. All the results mentioned in this paper are subject to the proposed computing environment; hence, the results can differ in k-fold crossvalidation in other computing environments. As mentioned earlier, we did not remove any artifact from the EEG data, and the results may vary with more preprocessing of the data. Finally, we compared our results with state-of-the-art methods proposed for the sparse representation-based classification, and the results may differ from traditional classification models.

4.5. Conclusions

In this study, we aimed at developing a novel SRC-based classification framework that is suitable for real-time classification. We have investigated and compared our proposed methodology with different features, epoch lengths, selected channels and epoch lengths, classification models, and state-of-the-art studies. The proposed methodology uses sparse wavelet features (energy and entropy) and 1D-ConvResNet and outperforms state-of-the-art models for SRC-based classification.

Author Contributions: Conceptualization, H.G. and V.M.; methodology, H.G.; software, H.G.; validation, V.M. and H.G.; formal analysis, H.G.; investigation, H.G.; resources, V.M.; data curation, H.G.; writing—original draft preparation, H.G.; writing—review and editing, H.G. and V.M.; visualization, H.G.; supervision, V.M. All authors have read and agreed to the published version of the manuscript.

Funding: There is no funding for the present work.

Institutional Review Board Statement: The dataset is already available in the OpenNeuro Database. We did not do any data collection from human subjects for this paper.

Data Availability Statement: The dataset is available in OpenNeuro Database: <https://openneuro.org/datasets/ds003655/versions/1.0.0> (accessed on 28 February 2023).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Gurve, D.; Delisle-Rodriguez, D.; Bastos-Filho, T.; Krishnan, S. Trends in Compressive Sensing for EEG Signal Processing Applications. *Sensors* **2020**, *20*, 3703. [CrossRef] [PubMed]
2. Jin, J.; Liu, C.; Daly, I.; Miao, Y.; Li, S.; Wang, X.; Cichocki, A. Bispectrum-Based Channel Selection for Motor Imagery Based Brain-Computer Interfacing. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2020**, *28*, 2153–2163. [CrossRef] [PubMed]
3. Lee, B.-H.; Jeong, J.-H.; Shim, K.-H.; Lee, S.-W. Classification of High-Dimensional Motor Imagery Tasks Based on an End-to-End Role Assigned Convolutional Neural Network. In Proceedings of the ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 1359–1363. [CrossRef]
4. Raza, H.; Rathee, D.; Zhou, S.-M.; Cecotti, H.; Prasad, G. Covariate shift estimation based adaptive ensemble learning for handling non-stationarity in motor imagery related EEG-based brain-computer interface. *Neurocomputing* **2019**, *343*, 154–166. [CrossRef] [PubMed]
5. Leelaarporn, P.; Wachiraphan, P.; Kaewlee, T.; Udsa, T.; Chaisaen, R.; Choksatchawathi, T.; Laosirirat, R.; Lakhan, P.; Natnithikarat, P.; Thanontip, K.; et al. Sensor-Driven Achieving of Smart Living: A Review. *IEEE Sens. J.* **2021**, *21*, 10369–10391. [CrossRef]
6. Pilastrri, A.L.; Tavares, J.M.R. Reconstruction Algorithms in Compressive Sensing: An Overview. In Proceedings of the 11th Edition of the Doctoral Symposium in Informatics Engineering (DSIE-16), Porto, Portugal, 3 February 2016; pp. 127–137.

7. Dittthapron, A.; Banluesombatkul, N.; Kettrat, S.; Chuangsuwanich, E.; Wilaiprasitporn, T. Universal Joint Feature Extraction for P300 EEG Classification Using Multi-Task Autoencoder. *IEEE Access* **2019**, *7*, 68415–68428. [[CrossRef](#)]
8. Shrivastwa, R.R.; Pudi, V.; Duo, C.; So, R. A brain-computer interface framework based on compressive sensing and deep learning. *IEEE Consum. Electron. Mag.* **2020**, *9*, 90–96. [[CrossRef](#)]
9. Gao, G.; Shang, L.; Xiong, K.; Fang, J.; Zhang, C.; Gu, X. EEG Classification Based on Sparse Representation and Deep Learning. *Neuroquantology* **2018**, *16*. [[CrossRef](#)]
10. Sreeja, S.R.; Samanta, D. Classification of multiclass motor imagery EEG signal using sparsity approach. *Neurocomputing* **2019**, *368*, 133–145. [[CrossRef](#)]
11. Zeng, K.; Yan, J.; Wang, Y.; Sik, A.; Ouyang, G.; Li, X. Automatic detection of absence seizures with compressive sensing EEG. *Neurocomputing* **2016**, *171*, 497–502. [[CrossRef](#)]
12. Chaisaen, R.; Autthasan, P.; Mingchinda, N.; Leelaarporn, P.; Kunaseth, N.; Tammajarung, S.; Manoonpong, P.; Mukhopadhyay, S.C.; Wilaiprasitporn, T. Decoding EEG Rhythms During Action Observation, Motor Imagery, and Execution for Standing and Sitting. *IEEE Sens. J.* **2020**, *20*, 13776–13786. [[CrossRef](#)]
13. Wilaiprasitporn, T.; Dittthapron, A.; Matchaparn, K.; Tongbuasirilai, T.; Banluesombatkul, N.; Chuangsuwanich, E. Affective EEG-Based Person Identification Using the Deep Learning Approach. *IEEE Trans. Cogn. Dev. Syst.* **2019**, *12*, 486–496. [[CrossRef](#)]
14. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778. [[CrossRef](#)]
15. Lema-Condo, E.L.; Bueno-Palomeque, F.L.; Castro-Villalobos, S.E.; Ordonez-Morales, E.F.; Serpa-Andrade, L.J. Comparison of Wavelet Transform Symlets (2-10) and Daubechies (2-10) for an Electroencephalographic Signal Analysis. In Proceedings of the 2017 IEEE XXIV International Conference on Electronics, Electrical Engineering and Computing (INTERCON), Cusco, Peru, 15–18 August 2017; pp. 1–4. [[CrossRef](#)]
16. Gowri, M.S.G.; Raj, D.P.P. EEG Feature extraction using Daubechies Wavelet and Classification using Neural Network. *Int. J. Comput. Sci. Eng.* **2019**, *7*, 792–799. [[CrossRef](#)]
17. Jacob, J.E.; Nair, G.K.; Iype, T.; Cherian, A. Diagnosis of Encephalopathy Based on Energies of EEG Subbands Using Discrete Wavelet Transform and Support Vector Machine. *Neurol. Res. Int.* **2018**, *2018*, 3209–3223. [[CrossRef](#)] [[PubMed](#)]
18. Gajbhiye, P.; Mingchinda, N.; Chen, W.; Mukhopadhyay, S.C.; Wilaiprasitporn, T.; Tripathy, R.K. Wavelet Domain Optimized Savitzky–Golay Filter for the Removal of Motion Artifacts From EEG Recordings. *IEEE Trans. Instrum. Meas.* **2020**, *70*, 1–11. [[CrossRef](#)]
19. Mao, Y.; Huiyang, Z.; Gui, X. *Exploring Deep Neural Networks for Branch Prediction*; ECE Department, NC University: Raleigh, NC, USA, 2018; pp. 1–7.
20. Peng, D.; Liu, Z.; Wang, H.; Qin, Y.; Jia, L. A Novel Deeper One-Dimensional CNN With Residual Learning for Fault Diagnosis of Wheelset Bearings in High-Speed Trains. *IEEE Access* **2018**, *7*, 10278–10293. [[CrossRef](#)]
21. Srivastava, R.K.; Greff, K.; Schmidhuber, J. Training very deep networks. *Adv. Neural Inf. Process. Syst.* **2015**, *28*.
22. Goldberger, A.L.; Amaral, L.A.N.; Glass, L.; Hausdorff, J.M.; Ivanov, P.C.; Mark, R.G.; Mietus, J.E.; Moody, G.B.; Peng, C.-K.; Stanley, H.E. PhysioBank, PhysioToolkit, and PhysioNet: Components of a New Research Resource for Complex Physiologic Signals. *Circulation* **2000**, *101*, E215–E220. [[CrossRef](#)]
23. Schalk, G.; McFarland, D.; Hinterberger, T.; Birbaumer, N.; Wolpaw, J. BCI2000: A General-Purpose Brain-Computer Interface (BCI) System. *IEEE Trans. Biomed. Eng.* **2004**, *51*, 1034–1043. [[CrossRef](#)]
24. Sreeja, S.R.; Himanshu; Samanta, D. Distance-based weighted sparse representation to classify motor imagery EEG signals for BCI applications. *Multimed. Tools Appl.* **2020**, *79*, 13775–13793. [[CrossRef](#)]
25. Wang, H.; Tang, C.; Xu, T.; Li, T.; Xu, L.; Yue, H.; Chen, P.; Li, J.; Bezerianos, A. An Approach of One-vs-Rest Filter Bank Common Spatial Pattern and Spiking Neural Networks for Multiple Motor Imagery Decoding. *IEEE Access* **2020**, *8*, 86850–86861. [[CrossRef](#)]
26. Huang, J.-S.; Chen, B.-Q.; Zeng, N.-Y.; Cao, X.-C.; Li, Y. Accurate classification of ECG arrhythmia using MOWPT enhanced fast compression deep learning networks. *J. Ambient. Intell. Humaniz. Comput.* **2020**. [[CrossRef](#)]
27. Hayou, S.; Doucet, A.; Rousseau, J. On the Impact of the Activation Function on Deep Neural Networks Training. In Proceedings of the 36th International Conference on Machine Learning, ICML 2019, Long Beach, CA, USA, 9–15 June 2019; pp. 4746–4754.
28. Ioffe, S.; Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In Proceedings of the International Conference on Machine Learning, Lille, France, 6–11 July 2015.
29. Zhang, Y.; Nam, C.S.; Zhou, G.; Jin, J.; Wang, X.; Cichocki, A. Temporally Constrained Sparse Group Spatial Patterns for Motor Imagery BCI. *IEEE Trans. Cybern.* **2018**, *49*, 3322–3332. [[CrossRef](#)]
30. Jin, J.; Miao, Y.; Daly, I.; Zuo, C.; Hu, D.; Cichocki, A. Correlation-based channel selection and regularized feature optimization for MI-based BCI. *Neural Netw.* **2019**, *118*, 262–270. [[CrossRef](#)]
31. Bagh, N.; Machireddy, R. Detection of Motor Imagery Movements Based on the Features of Phase Space Reconstruction. In Proceedings of the 2019 9th International IEEE/EMBS Conference on Neural Engineering (NER), San Francisco, CA, USA, 20–23 March 2019; pp. 916–919. [[CrossRef](#)]
32. Blankertz, B.; Muller, K.-R.; Krusienski, D.; Schalk, G.; Wolpaw, J.; Schlogl, A.; Pfurtscheller, G.; Millan, J.; Schroder, M.; Birbaumer, N. The BCI competition III: Validating alternative approaches to actual BCI problems. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2006**, *14*, 153–159. [[CrossRef](#)]

33. Wang, X.; Hersche, M.; Tömekce, B.; Kaya, B.; Magno, M.; Benini, L. An Accurate EEGNet-Based Motor-Imagery Brain–Computer Interface for Low-Power Edge Computing. In Proceedings of the 2020 IEEE International Symposium on Medical Measurements and Applications (MeMeA), Bari, Italy, 1 June–1 July 2020.
34. Lawhern, V.J.; Solon, A.J.; Waytowich, N.R.; Gordon, S.M.; Hung, C.P.; Lance, B.J. EEGNet: A compact convolutional neural network for EEG-based brain–computer interfaces. *J. Neural Eng.* **2018**, *15*, 056013. [[CrossRef](#)]
35. She, Q.; Chen, K.; Ma, Y.; Nguyen, T.; Zhang, Y. Sparse Representation-Based Extreme Learning Machine for Motor Imagery EEG Classification. *Comput. Intell. Neurosci.* **2018**, *2018*, 1–9. [[CrossRef](#)]
36. Deng, X.; Zhang, B.; Yu, N.; Liu, K.; Sun, K. Advanced TSGL-EEGNet for Motor Imagery EEG-Based Brain–Computer Interfaces. *IEEE Access* **2021**, *9*, 25118–25130. [[CrossRef](#)]
37. Singh, A.; Lal, S.; Guesgen, H.W. Small Sample Motor Imagery Classification Using Regularized Riemannian Features. *IEEE Access* **2019**, *7*, 46858–46869. [[CrossRef](#)]
38. Kwon, O.-Y.; Lee, M.-H.; Guan, C.; Lee, S.-W. Subject-Independent Brain–Computer Interfaces Based on Deep Convolutional Neural Networks. *IEEE Trans. Neural Netw. Learn. Syst.* **2019**, *31*, 3839–3852. [[CrossRef](#)]
39. Banluesombatkul, N.; Ouppaphan, P.; Leelaarporn, P.; Lakhan, P.; Chaitusaney, B.; Jaimchariyatam, N.; Chuangsuwanich, E.; Chen, W.; Phan, H.; Dilokthanakul, N.; et al. MetaSleepLearner: A Pilot Study on Fast Adaptation of Bio-Signals-Based Sleep Stage Classifier to New Individual Subject Using Meta-Learning. *IEEE J. Biomed. Health Inform.* **2021**, *25*, 1949–1963. [[CrossRef](#)]
40. Fadel, W.; Kollod, C.; Wahdow, M.; Ibrahim, Y.; Ulbert, I. Multi-Class Classification of Motor Imagery EEG Signals Using Image-Based Deep Recurrent Convolutional Neural Network. In Proceedings of the 2020 8th International Winter Conference on Brain-Computer Interface (BCI), Gangwon, Republic of Korea, 26–28 February 2020; pp. 1–4. [[CrossRef](#)]
41. Cai, S.; Shu, Y.; Chen, G.; Ooi, B.C.; Wang, W.; Zhang, M. Effective and efficient dropout for deep convolutional neural networks. *arXiv* **2019**, arXiv:1904.03392.
42. Roots, K.; Muhammad, Y.; Muhammad, N. Fusion Convolutional Neural Network for Cross-Subject EEG Motor Imagery Classification. *Computers* **2020**, *9*, 72. [[CrossRef](#)]
43. Dose, H.; Moller, J.S.; Puthusserypady, S.; Iversen, H.K. A Deep Learning MI-EEG Classification Model for BCIs. In Proceedings of the 26th European Signal Processing Conference, Rome, Italy, 3–7 September 2018; pp. 1676–1679. [[CrossRef](#)]
44. Lun, X.; Yu, Z.; Chen, T.; Wang, F.; Hou, Y. A Simplified CNN Classification Method for MI-EEG via the Electrode Pairs Signals. *Front. Hum. Neurosci.* **2020**, *14*, 338. [[CrossRef](#)] [[PubMed](#)]
45. Mishra, A.; Thakkar, F.N.; Modi, C.; Kher, R. Selecting the Most Favorable Wavelet for Compressing ECG Signals Using Compressive Sensing Approach. In Proceedings of the 2012 International Conference on Communication Systems and Network Technologies, Rajkot, India, 11–13 May 2012; pp. 128–132. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.